

HDTV Video Server

Background of the InventionRelated Applications:

- 5 This application claims priority from copending United States provisional patent application serial no. 60/180,098 filed February 3, 2000.

1. Field of the Invention:

- 10 This invention relates to video production and processing, and more specifically to real-time processing RGB format, high definition television (HDTV) image files for HDTV production equipment.

2. Description of the Prior Art:

- 15 Born of feature film special effects roots, electronic cinematography (EC) is a product of computer graphics imaging (CGI) systems and software. The need to manipulate elemental imagery as individual computer files is inherent in digital feature film production. Principle photography for a feature film may produce as much as 20 hours of 24 frames per second
20 (fps) digital imagery. This is more than 17 million frames, each of which needs to be made available for CGI processing and dailies review. Iterative dailies review and compositing will produce many tens of hundreds of thousands of frames per day.

- 25 The costs associated with software conversion of digital HDTV frames into computer files are often hidden. Traditional conversions have relied upon software techniques and are processor and memory intensive. Each transversal of the production pipeline by a frame may entail many such

conversions. Every effort to reduce, or eliminate, the use of processor time for these conversions pays dividends many times over by freeing valuable processors for other more directly billable software tasks.

5 A Redundant-Array-of-Inexpensive-Disks (RAID) configuration is not used because RAID systems operate at diminished capacities and rates during failure modes. This is anathema to real-time playback and recording. Though the lack of redundant data seems a malady, in practice it is not really
10 so. In the event of a disk failure, the offending disk is replaced, and the frames re-edited to, or from, the reconstructed file system. This is far faster than the downtime encountered during RAID rebuilds. Additionally, RAID arrays must also calculate and write extra parity data to the
15 array. This can cause increased write times compared to non-RAID arrays. When transaction times in the milliseconds are important, an increase in access time can be too much, causing a loss of frames.

20 The native colorspace of HDTV is YCbCr; a 2/3 compressed colorspace that shares adjacent pixel color information, reducing the total storage size for each frame. Conventional CGI software generally uses linear RGB colorspaces that may require more than 3 times the storage of a YCbCr frame. Also, linear RGB colorspaces may require gamma correction to overcome
25 the gamma introduced by video equipment.

30 Some mechanism must exist to convert between HDTV's YCbCr and CGI's RGB formats. Conventional conversion techniques are software based, and while this works, it is time consuming and requires many processor-hours daily, and most certainly does not support real-time operation. What is needed is an

efficient method and apparatus to transfer and convert frames between HDTV equipment and CGI systems in real-time.

Summary of the Invention

5 The present invention may provide a real-time technique for processing YCbCr images into RGB format files. An HDTV video server according to the present invention translates YCbCr images to RGB12 image files and includes a several parallel memory paths and parallel storage devices to minimize
10 data bottlenecks.

 In another aspect of the present invention a method of real-time translation of YcbCr images into RGB images includes the steps of capturing a high-density video image in a first data format, compiling the high definition video image in a
15 second data format and writing the high definition video image as a stripped data file.

 In a still further aspect of the present invention, an HDTV video server according to the present invention includes means for translating the high definition video image in a
20 first data format to a high definition video image in a second data format, means for filtering the high definition video image to eliminate translation artifacts, means for correcting the high definition video image, means for packing the high definition video image in a second data format packing mode,
25 means for writing the high definition video image as a stripped data file, means for reading the stripped data file and compiling the high definition video image in the second data format, and means for providing the high definition video image in the second data format to a network or other device.

30 These and other features and advantages of this invention will become further apparent from the detailed description and

accompanying figures that follow. In the figures and description, numerals indicate the various features of the invention, like numerals referring to like features throughout both the drawings and the description.

5

Brief Description of the Drawings

Fig. 1 is a table listing data rates for various formats and frame rates.

Fig. 2A is a table showing bit packing for YCbCr8 format.

10 Fig. 2B is a table showing bit packing for YCbCr10 format.

Fig. 2C is a table showing bit packing for RGB8 format.

Fig. 2D is a table showing bit packing for RGB10 format.

Fig. 2E is a table showing bit packing for RGB12 format.

15 Fig. 3 is a block diagram of a network incorporating a video server according to the present invention.

Fig. 4 is a block diagram of a video server according to the present invention.

Fig. 5 is a flow chart for frame buffer data processing according to the present invention.

20 Fig. 6 is a diagram of the card layout for an HDTV video server according to the present invention.

Detailed Description of the Preferred Embodiment(s)

25 Referring now to Fig. 1, overall data rates for various modes of packing and colorspace are outlined in table 10. These data rates are the actual image payload data rates only. Operating system and application I/O may further burden the

data rates shown. Transaction rates are defined as the frame rate to process individual frames, and at 24 fps, frames must be read or written every ~41 milli seconds (msecs).

Figures 2A-2F shows various forms of packing, or the manner in which bits are stuffed together to represent pixel data. Packing impacts storage size significantly. Referring now to Fig. 2A, eight bit YCbCr files require 2/3 the storage of the their 8 bit RGB cousins of Fig. 2C. Higher color depth RGB files such as RGB12 require even more bandwidth. The RGB12 packing mode of Fig. 2E is the currently preferred packing mode of the present invention. With 1920 pixels, 1080 lines, and 6 bytes per pixel, each frame requires more than 12.5 megabytes (MB) of storage and provides maximum color depth.

There are two basic methods of writing high data rate files according to the present invention: open and close per frame (OCF), and streaming. OCF has the advantage of creating individual computer files for each frame, but the disadvantage of dramatically increasing the transaction rate requirements. Single frame files must be opened, written/read, and closed in a timely fashion so that each and every frame is handled without loss or delay. The streaming approach opens and closes only a single file per shot, creating one very large file, storing or retrieving individual frames by offsetting into the file. Streaming is kinder to the file system because only one open and close is encountered per shot but produces single large files of collections of frames. At the desired RGB12 color depth, the streaming method produces files of ~18GB/minute.

A currently preferred embodiment of the present invention uses OCF methods for two reasons: 1) moving massive streamed

files over a network infrastructure is time consuming and problematic, and 2) extracting individual frames from a single massive streamed file requires further processing. A secondary downstream process must extract or insert frames to gain access to individual frames. These hindrances complicated the design goal of real-time access to frames.

For an RGB12 system according to the present invention, and considering data rates, transaction rates, and packing, at 24 fps, the data rate is ~300MB/sec., or about ~18GB/minute. Individual frames of ~12.5 MB must be stored or retrieved every 41.8 msec. Referring now to Fig. 4, image data 24 must also transit from frame buffer 28 into memory 30, and then from memory 30 to storage such as network storage 32, producing an aggregate bandwidth across computer bus 34 of twice the expected data rate. This results in a total bandwidth requirement of 600MB/sec.

Referring now to Fig. 3, a high-resolution video processor and capture device 20 according to the present invention, may be connected to network 22 as shown. Network 22 may include one or more users 36 and storage 32. Data 24 may be applied to high-resolution video processor and capture device 20 using interface 26. Memory 30 is provided as local memory. Control information 25 may be provided through serial port 38 which may include an appropriate converter such as converter 40 for conversion between RS-232 and RS-422.

To achieve these staggering data rates, careful attention must be given not only to the numbers of CPUs, disks, frame buffers, and memory sizes, but also to platform backplane design and avoidance of I/O bottlenecks.

Referring now to Fig. 4 the detail block diagram of high-resolution video processor and capture device 20 includes the following functional blocks:

- 4 node processor system 50 such as the SGI Origin 2000;
- 5 • Frame Buffer 28 such as the SGI XT-HD Frame Buffer Card (with outboard serial/parallel converters);
- Computer bus 34 or backplane such as the SGI XIO High-speed backplane;
- Multiple memory nodes 54, or frame buffers;
- Multiple Fiber Channel (FC) interface boards 52;
- 10 • "Stripped" multiple parallel disk drive storage sub-systems such as disk drive 55;
- RS422 control port 40.

In a currently preferred embodiment of the present invention, processor system 50 may be an SGI 4-node Origin 2000 platform, chosen for it's high-speed XIO bus 34. Frame buffer 28 may be an SGI XT-HD frame buffer suitable for its outstanding RGB conversion ability, especially in light of its capability to deliver the desired RGB12 packing mode. Multiple processor cards 51 provide more memory nodes 54 that enable parallel memory access. Fiber Channel interface boards 52 were chosen for their ease of use and high throughput. Other suitable components may be used.

The basic system is a bridge between two worlds. The two sides are split between computer file system side 20C, and television production side 20T. On computer side 20C, network connectivity provides for network access to individual frames of HDTV material as RGB files. Multiple protocols such as gigabit ethernet, serial HIPPI, and switched 100bT access may be used. On television production side 20T, high-resolution

video processor and capture device 20 behaves as an industry standard VTR, interfacing to existing edit environments exactly as any VTR would. A proprietary RS422 software daemon 42 provides edit system operation. The video input and output may be routed to appropriate sources or destinations. Serial port 38 handles RS422 communications via RS232 to RS422 level converter 40 and routed with data 24 to appropriate control devices such as control/editor 44.

Frame buffer 28 acquires and or transmits the standard HDTV data streams at the necessary frame rates, converting the frames to, or from, the RGB colorspace, and providing image correction. In a currently preferred embodiment of the present invention an SGI XT-HD frame buffer operates with parallel I/O only, so outboard converters 28I and 28O may be necessary to accommodate SMPTE 292 HD SDI signals.

Referring now to Fig. 5, a block diagram of the data flow within frame buffer 28 is shown. At step 80 frame buffer 28 buffers the input data stream 24I before passing it to conversion matrix for conversion to the RGB colorspace in step 82. The frame buffer of the present invention supports three matrices, SMPTE 274M, ITU-BT Rec.709 and ITU-BT Rec.601, any one of which is selectable at initialization time. Other suitable conversion techniques may also be used. At step 84, data are passed to 13-bit filter 48 where ringing and edge artifacts inherent in colorspace conversions are diminished. At step 86 data are passed to transform block 68 which uses look-up-table(LUT) or other suitable techniques to map input values to new output values. It is during step 86 that gamma corrections, or other more esoteric mappings, may occur. Data 24I are passed to the packing block 70 at step 88, where data 24I is formatted for the requested packing mode such as RGB8,

RGB10, or RGB12. The preferred packing mode is RGB12. At step 90, the data are transferred via DMA block 72 into or out of frame buffer memory 28M.

Throughput in a system such as system 12 is ultimately
5 determined by the design of the backplane or computer bus 34.
It must be capable of consistently transferring the total data
rates shown in Figure 1, without blocking or dropping data. In
a currently preferred embodiment of the present invention, a
SGI Origin XIO backplane adequately supports the RGB12 packing
10 mode, any other suitable equipment may also be used. As shown
in Fig. 1, using RGB12 at 24 fps, the bandwidth requirement is
~300MB/second. This means an aggregate throughput of
~600MB/sec., since each frame must transit the bus twice, once
into memory, and thence to the storage sub-systems, This
15 600MB/sec. rate is just under the published maximum threshold
of operation of the XIO bus.

The application software provides frame buffers to receive
frames from the storage sub-system or the HD frame buffer
depending on whether frames are playing or recording. These
20 buffers are evenly split across the available CPU nodes
providing simultaneous parallel paths for data flow. This
modular and adaptable design assures that no single bottleneck
exists that will completely dominate the I/O process, and can
be configured for optimal bandwidth and transaction rates for
25 any given color depth. This mitigates the interaction of
bandwidth, number of FC ports, and transaction rates.

Referring now to Fig. 6, the relative layout of the cards
of the present invention is shown. In high-speed designs
careful attention must be paid to actually getting the
30 throughput required to support the needed data rates. For

instance, in this design, correct placement of the FC cards 53 in the chassis slots 55 with respect to frame buffer card 57 is critical. Incorrectly placing the FC cards will cause I/O imbalance, and the transfer rates at any single node may exceed the maximum and the system will produce tearing and flashing video.

The disk storage sub-system or memory 30 is the repository of inbound or outbound frames, data 24. It must have sufficient capacity to store the number of frames expected and it must be capable of the required bandwidth at the transaction rates of 24 and 30 fps. It must also be flexible and cost effective.

No single disk drive can support these data rates. Therefore multiple disk drives or memory nodes such as memory node 54 are "stripped", or sequentially written, with each frame. The stripping process may be optimized using software 56 such as the SGI XFS file system. A single frame of data 24 is stripped across the array of disks 54, each disk shouldering its portion of the overall data in parallel with the others. With the appropriate number of drives, the tremendous data rate associated with RGB12 HDTV frame files can be accommodated. In a currently preferred embodiment of the present invention 32 50GB hard drives or memory nodes 54 are used.

To optimize frame transfer between processor system 50 and memory 30, 8 FC pipes 58 are used, and at a data rate of ~300 MB/sec., each stripe is responsible for storing about ~37.5 MB of data per second. Each FC pipe 58 is connected to 4 50GB drives, for a total storage of 1.6TB. This provides for approximately 1.5 hours of RGB12 storage at 24 fps.

Another feature endemic to disk drives that must be mitigated is thermal recalibration. This process keeps heads properly aligned to data tracks. During thermal recalibration, disk I/O activity is suspended, producing image freezes or loss of frames. This is not a desirable feature and it is vital to eliminate or hide the thermal recalibration process. Many manufacturers hide the thermal recalibration between data access, but many do not. To achieve the high data throughput of the present invention it is necessary to use disk drives such as memory node 54 that perform thermal recalibration between data access.

Another element of the storage sub-systems that effects performance is file system logging. Using separate disk drives such as admin drive 60 for file log 62 prevents logging operations from interfering with realtime frame I/O. Journalizing file systems such as SGI's XFS write log files to maintain meta-data. By default, XFS uses a log on the same disks as the file system it is managing, but may optionally locate the log on entirely separate disks, any similarly suitable file system may also be used.

In a currently preferred embodiment of the present invention, the software portion of the system includes operating system 64 and it's libraries, and custom UNIX server daemon processes 66. A daemon is a process that runs in the background and performs a specified operation at predefined times or in response to certain events. The term daemon is a UNIX term, though many other operating systems provide support for daemons, though they're sometimes called other names. Windows, for example, refers to daemons and System Agents and services. Any suitable operating system may be used.

SGI's IRIX 6.5.5 is the OS used according to the present invention. It's dmedia library directly supports frame buffer 28 with drivers that provide the appropriate HDTV signals and packing modes, RGB conversions, look up table support, and
5 frame buffer distribution. Daemon processes 66 according to the present invention use the SGI dmedia libraries to build HDTV video server 20.

Main daemon process 80 consists of multiple children processes, that run concurrently to execute user commands,
10 RS422 serial control, frame transfers, and associated tasks. There are two main processes, the server itself, and an RS422 edit control module. With each request from a user, an additional child process is forked off to handle the actual realtime I/O of frames. Locking daemons prevent simultaneous
15 server access. Routing daemons provide signal routing control that relieves operators from manually routing signals. Database daemons provide global shot and device control information. Each of the software processes makes extensive use of logging for troubleshooting and diagnosis, and cost
20 accounting and usage reports.

The main server daemon is responsible for initializing the appropriate hardware and system resources. It runs at a high system priority to minimize contention by other system processes. After starting, it immediately spawns the RS422
25 child process, and then sits around waiting for user commands. User commands are received via a TCP/IP socket connection and can be transmitted from a variety of user interfaces. Commands and parameters may be sent from command lines, from shell scripts, from web based interfaces, and from high-performance
30 GUI interfaces.

The RS422 daemon child process 42 is responsible for responding to industry standard editing control commands. Without process 42, editing system operation would be impossible. Using the SGI tserialio library, master/slave RS422 interface, interface 41, was constructed. Daemon process 42 maintains a virtual VTR state-machine that provides the control information necessary for all other daemons. The RS422 daemon runs continuously, providing for "always-on" status and timecode for edit controllers. Even if no playback or record command has been issued by the server daemon, the RS422 daemon is responsive. It behaves as if there were no "tape" in the "deck". To shut down operations, issuing the eject command causes the child frame I/O server process to shutdown, and the main daemon to become available for further user commands.

Having now described the invention in accordance with the requirements of the patent statutes, those skilled in this art will understand how to make changes and modifications in the present invention to meet their specific requirements or conditions. Such changes and modifications may be made without departing from the scope and spirit of the invention as set forth in the following claims.